EXPLORING GENERALIZATION OF NON-CONTRASTIVE SELF-SUPERVISED LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Contrastive learning have recently produced results comparable to the state-of-the-art supervised models. Non-contrastive methods do not use negative samples, but separate samples of different classes by explicitly or implicitly optimizing the representation space. Although we have some understanding of the core of the non-contrastive learning method, theoretical analysis of its generalization performance is still missing. Thus we present a theoretical analysis of generalizability of non-contrastive models. We focus on the inter-class distance, show how non-contrastive methods increase the inter-class distance, and how the distance affects the generalization performance of the model. We find that the generalization of non-contrastive methods is affected by the output dimension and the number of latent classes. Models with much fewer dimensions than the number of latent classes are not sufficient to generalize well. We demonstrate our findings through experiments on the CIFAR dataset.

1 INTRODUCTION

Self-Supervised Learning (SSL) is gaining popularity as a result of its competitive performance comparing to supervised learning, while is free of costly labels. Among SSL models, contrastive learning has attracted great attention due to its strong generalization performance and wide range of applications. Contrastive learning generates multiple augmented views from samples, and treats the views generated from the same sample with the same label, whereas the views from different samples have different labels. Contrastive models have been widely used in various of scenarios including but not limited to graph representation (You et al., 2020; Zhu et al., 2020), computer vision (Chuang et al., 2020; Tian et al., 2020a) and natural language processing (Saeed et al., 2021; Iter et al., 2020; Chi et al., 2020).

Typical contrastive models like MoCo (He et al., 2020)and SimCLR (Chen et al., 2020) use InfoNCE loss to attract positive pairs and push negative pairs. However a negative sample of contrastive learning can be false negative (Huynh et al., 2022). Views of two samples with the same latent will be treated as negative pairs. This can hurt the performance of the model. There are many works trying to avoid this disadvantage (Huynh et al., 2022; Kalantidis et al., 2020; Kim et al., 2021). BYOL (Grill et al., 2020) and SimSiam (Chen & He, 2020) uses an asymmetric network structure, cancel the use of negative samples, and surprisingly achieves comparable performance. Tian et al. (2021); Jing et al. (2021) analyse the dynamics of features and networks in BYOL and SimSiam and explore why they perform well without falling into trivial solutions. However, studies that quantitatively investigate their generalization performance have yet to emerge.

In this paper, we explore the generalizability of non-contrastive learning. Under appropriate assumptions, we prove that the cross-correlation matrix of output features constrains the upper bound of generalization error rate. The closer the matrix is to the identity matrix, the smaller this upper bound becomes. Moreover, we demonstrate that the cross-correlation matrix is optimized by non-contrastive methods during training.

The sections of this paper are arranged as follow: In Section 2 we review contrastive learning methods and the theoretical analysis of contrastive learning. Then in Section 3 we formulate the generalization problem and agree on some notation. In Section 4 the error rate theorem is proposed. It illustrates how the cross-correlation matrix and the dimensionality of the representation space can limit the upper bound on the error rate. Then we shows that SimSiam (Chen & He, 2020) and Barlow

Twins (Zbontar et al., 2021) optimize the cross-correlation matrix during training. At last in Section 5 we conduct experiments on CIFAR-10 and CIFAR-100 to verify the effect of feature dimension and number of classes on error rate in Section 4.

2 Related Works

2.1 Contrastive Self-Supervised Learning

Self-supervised learning methods strive to learn representations using well-designed pretext tasks that do not require the use of expensive labels. As a kind of self-supervised learning method, contrastive learning uses data augmentation to generate multiple augmented samples (views) from an original sample, and treats the augmented samples generated by the same sample with the same label, whereas the augmented samples from different samples have different labels. Recent methods such as MoCo (He et al., 2020)and SimCLR (Chen et al., 2020) have produced results comparable to the state-of-the-art supervised method on ImageNet (Deng et al., 2009) dataset. They use the InfoNCE loss (Van den Oord et al., 2018) to pull positive sample pair together while pushing away negative samples, i.e., those augmented views from different samples. However views from different sample can have the same latent label. Treating them as negative samples will hurts model performance. To overcome this shortcoming, Hu et al. (2021); Xiao et al. (2020) carefully design the negative sampling process. Some works like BYOL (Grill et al., 2020), SimSiam (Chen & He, 2020) and Barlow Twins (Zbontar et al., 2021) directly created models that do not use negative samples at all. These contrastive models free of negative samples are also called non-contrastive models.

2.2 Theoretical Analysis for Contrastive Learning

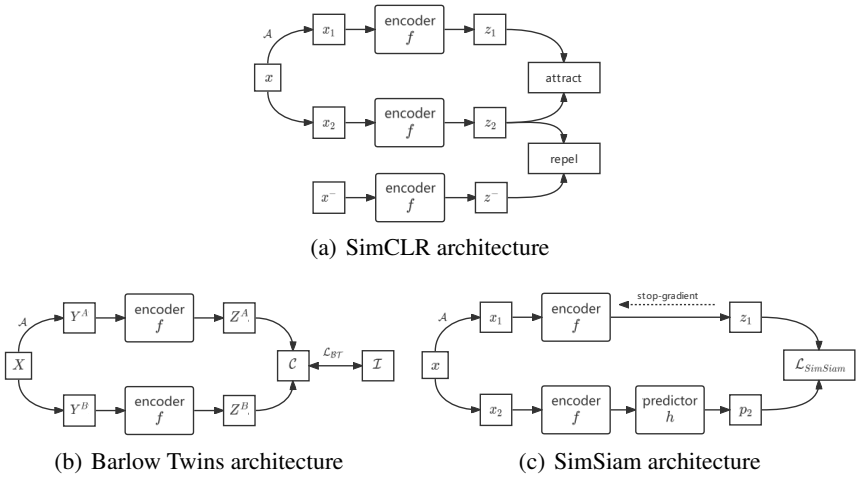
Despite the fact that contrastive learning has shown to be effective in learning usable representations and has outperformed supervised learning in key downstream transfer learning benchmarks (He et al., 2020), their underlying mechanism remain opaque and poorly understood. Arora et al. (2019) provides a theoretical analysis on the explanation why the learned features via contrastive learning are useful for downstream tasks. Tian et al. (2021); Jing et al. (2021) give theoretical analysis of dynamics of non-contrastive methods and investigate how these models avoid trivial solutions. Huang et al. (2021) studies the generalization property of SimCLR and Barlow Twins, however, methods like BYOL with special optimization procedures are not considered. Tosh et al. (2021) study the generalization performance in the aspect of the mutual information. Tian et al. (2020b) propose the optimal views for contrastive learning that reserve relevant task information to ensure the mutual information among all views are task-relevant.

Research above explored the learning mechanism of contrastive learning and proposed a number of methods to modify the generalization performance of the model using loss function, however, most of which are based on InfoNCE object model with both positive and negative samples. Non-contrastive method with no negative samples can not be completely explained by theorems based on mutual information, and the analysis on its generalization performance also remains open. Our study focus on this open problem, exploring the factors that impact on self-supervised learning generalization performance, revealing how these factors affect the generalization error rate, and give out the upper bound of the non-contrastive model error rate.

3 Problem Formulation

In this section we formulate the loss of contrastive methods and the generalization error rate in preparation for the analysis in the next section.

Contrastive learning treats views of the same sample as having the same label. Given an underlying distribution $\mathcal{D} = \mathcal{C}_1 \cup \mathcal{C}_2 \cup \cdots \cup \mathcal{C}_K$. $\mathcal{C}_i, i \in [K]$ are disjoint latent classes. Each sample x is independently and identically distributed (i.i.d.) according to $\mathcal{D}$ and belongs to a unique $\mathcal{C}_i$. An augmentation $\mathcal{A}$ generate different views $\hat{x}$ of x. We denote $\mathcal{A}(x)$ and $\mathcal{A}(S)$ as the set of distorted views of x and sample set $S = (x_1, \ldots, x_m) \sim \mathcal{D}$. Typical models (Chen et al., 2020; He et al., 2020) learn an effective encoder f through pulling augmented views of the same sample together



(a) SimCLR architecture

(b) Barlow Twins architecture

(c) SimSiam architecture

Figure 1: The architectures of contrastive learning models.

and push views of all others away. They use the InfoNCE loss (Van den Oord et al., 2018):

$$\mathcal{L}_{\text{InfoNCE}} = -\mathbb{E}_{x\sim\mathcal{D},\hat{x}_1,\hat{x}_2\in\mathcal{A}(x)} \log \frac{\exp(f(\hat{x}_1)^T, f(\hat{x}_2))}{\exp(f(\hat{x}_1)^T, f(\hat{x}_2)) + \sum_{x^-\in\mathcal{A}(S)} \exp(f(\hat{x}_1)^T, f(\hat{x}^-))} \tag{1}$$

where $\hat{x}^- \in \mathcal{A}(S)$ are views from negative samples excluding x. The architecture of SimCLR (Chen et al., 2020) who use $\mathcal{L}_{\text{InfoNCE}}$ is shown in Fig. 1(a).

However, since we do not know the latent class of samples during training, some false negative samples will inevitably appear (Huynh et al., 2022). InfoNCE loss will push one view away from its false negative samples, that is, the negative samples that actually have the same latent label, which will have a negative impact on the model.

Non-contrastive learning models are contrastive methods without using negative samples. They either explicitly forces the cross-correlation matrix between positive samples close to the identity matrix (Barlow Twins (Zbontar et al., 2021), as shown in Fig. 1(b)), or makes use of asymmetric networks (BYOL (Grill et al., 2020), SimSiam (Chen & He, 2020), as shown in Fig. 1(c)) in order to avoid trivial constant features. The loss function of the former is:

$$\mathcal{L}_{\mathcal{BT}} = \sum_{i\in[d]} (1 - C_{ii})^2 + \lambda \sum_{i\in[d]} \sum_{j\neq i} C_{ij}^2, \tag{2}$$

where d is the dimension of the encoder outputs, $C = \mathbb{E}_{x\sim\mathcal{D},\hat{x}_1,\hat{x}_2\in\mathcal{A}(x)}(f(\hat{x}_1)f(\hat{x}_2)^T)$ is the cross-correlation matrix computed between the outputs of the two identical networks along the batch dimension and λ is a postive trade off parameter.

The latter is represented by SimSiam, whose loss function is:

$$\mathcal{L}_{SimSiam} = -\mathbb{E}_{x\sim\mathcal{D},\hat{x}_1,\hat{x}_2\in\mathcal{A}(x)}[\text{sim}(h(f(\hat{x}_1)), \text{sg}(f(\hat{x}_2)))], \tag{3}$$

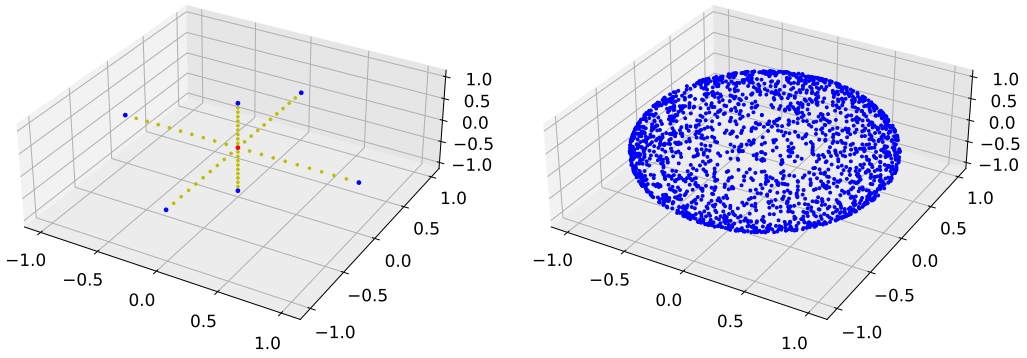
where $\text{sim}(\mathbf{x}, \mathbf{y}) = \frac{\mathbf{x}^T\mathbf{y}}{\|\mathbf{x}\|\|\mathbf{y}\|}$ is the cosine similarity between $(\mathbf{x}, \mathbf{y})$, h is the predictor layer, often a multi-layer perceptron. sg is the stop-gradient operator, which means the backpropagation does not go through this path.

As for the generalization, given an encoder f, we consider a nearest neighbor classifier $G : \mathcal{D} \to [K]$:

$$G(\hat{x}; f) = \arg\min_{k\in[K]} \|f(\hat{x}) - \mu_k\|, \tag{4}$$

where $\mu_k := \mathbb{E}_{x\sim\mathcal{C}_k,\hat{x}_1\in\mathcal{A}(x)}[f(\hat{x}_1)]$ is the mean of output features for samples in latent class $\mathcal{C}_k$. The classifier maps each $x \sim \mathcal{D}$ to a class $\mathcal{C}_i$. Here we denote $\mathcal{C}(x)$ as the real latent class of $x \sim \mathcal{D}$, the generalization error rate can be formulated as

$$Er(f) = \mathbb{P}[G(\hat{x}; f) \neq \mathcal{C}(x),\ \forall x \sim \mathcal{D},\ \forall \hat{x} \in \mathcal{A}(x)]. \tag{5}$$



(a) Distributions minimizing $\mathcal{L}_{cor}$ (b) Distributions minimizing $\mathcal{L}_{uniform}$

Figure 2: Distributions which minimize (a)$\mathcal{L}_{cor}$ and (b)$\mathcal{L}_{uniform}$. In Fig. (a), the red point is the origin of the coordinates, and the yellow points are the auxiliary points, which represents the relationship between the distribution (blue point) and the origin. It means that as long as the points fall on a set of orthonormal basis in the space on average, $\mathcal{L}_{cor}$ can be minimized, which is totally different from Fig. (b)

The generalization properties of $\mathcal{L}_{\text{InfoNCE}}$ have been well studied (Huang et al., 2021). In the following section we focus on Barlow Twins and SimSiam. Exploring the relationship between their optimization mechanism and generalization error rate.

4 Generalization Analysis

4.1 Error Rate Theorem

In this section, we study the generalization property of non-contrastive methods, and how the models improves generalization by optimizing the statistical characteristics of the output features.

Let d be the dimension of output features. For simplicity, we assume $\|f(x)\|^2 = r^2 = d$. Which means features $f(x)$ are scattered on a sphere with radius r. We assume that views of the same latent class are close to each other and never transfer to another class, formally:

$$\exists \varepsilon > 0,\ \forall S_i \sim \mathcal{C}_i,\ \forall \hat{x}_1, \hat{x}_2 \in \mathcal{A}(S_i),\ i \in [K],\ \|f(\hat{x}_1) - f(\hat{x}_2)\| < r\varepsilon, \tag{6}$$

$$\forall S_i \sim \mathcal{C}_i,\ \forall S_j \sim \mathcal{C}_j,\ i \neq j, \mathcal{A}(S_i) \cap \mathcal{A}(S_j) = \emptyset \tag{7}$$

Note the we suppose the encoder f is Lipschitz continuous so that $\hat{x}_1$ and $\hat{x}_2$ close to each other infers $f(\hat{x}_1)$ and $f(\hat{x}_2)$ are close.

In the next section, we will show that Barlow Twins and SimSiam optimize the following norm:

$$\mathcal{L}_{cor}(f) = \|\mathbb{E}_{x\sim\mathcal{D},\hat{x}\in\mathcal{A}(x)}[f(\hat{x})f(\hat{x})^T] - I_d\|^2, \tag{8}$$

rather than the Gaussian potential based uniformity(Wang & Isola, 2020):

$$\mathcal{L}_{uniform}(p;t) = \log \mathbb{E}_{x\sim\mathcal{D},\hat{x}_1,\hat{x}_2\in\mathcal{A}(x)} \left[G_t(f(\hat{x}_1), f(\hat{x}_2))\right],\ t > 0, \tag{9}$$

where $G_t(f(\hat{x}_1), f(\hat{x}_2)) = \exp(-t\|f(\hat{x}_1) - f(\hat{x}_2)\|_2^2)$ is the Gaussian potential kernel (Cohn & Kumar, 2007). These two kinds of optimization lead to different distribution patterns.

The distribution generated by $\mathcal{L}_{cor}(f)$ does not guarantee that each class of samples is separated from other classes. So the generalization of the model depends on the randomness in training and the dimension of the entire space and the number of latent classes. In this section, we first investigate when the model is guaranteed not to misclassify samples, and then derive an upper bound on the classification error rate in relation to $\mathcal{L}_{cor}$, d, and K.

As the assumption that views belong to the same latent class are close to each other, it is easy to think that as long as the inter-class distance between two classes is large enough, they will not be misclassified from each other. So we have the following theorem of a pair of latent classes. (See appendix for the proof)

Theorem 1 *If* $\mu_i^T \mu_j < r^2(1 - 4\varepsilon + \varepsilon^2), S_i \sim \mathcal{C}_i, S_j \sim \mathcal{C}_j$, *then*

$$\forall \hat{x}_1 \in \mathcal{A}(S_i),\ \forall \hat{x}_2 \in \mathcal{A}(S_j), G(\hat{x}_1; f) \neq j,\ G(\hat{x}_2; f) \neq i.$$

This theorem states that as long as μ_i and μ_j are far enough apart, views belong to class C_i and C_j will not be misclassified into each other's classes. This inspires us to study the constraints on the inter-class distance between latent classes.

Let $\Lambda = \mathbb{E}_{x\sim\mathcal{D},\hat{x}_1\in\mathcal{A}(x)}[f(\hat{x}_1)f(\hat{x}_1)^T]$. In the next section, we will show that non-contrastive losses $\mathcal{L}_{\mathcal{BT}}$ and $\mathcal{L}_{\text{SimSiam}}$ minimize $\mathcal{L}_{cor} = \|\Lambda - I_d\|^2$ explicitly and implicitly, respectively. Before that, here we prove that $\|\Lambda - I_d\|^2$ can constrains $\mu_i^T \mu_j$ and thus affects the error rate. (See appendix for the proof)

Theorem 2 *Let* $p_i = \mathbb{E}_{x\sim\mathcal{C}_k,\hat{x}\in\mathcal{A}(x)}[\mathbb{1}_{[\hat{x}\in\mathcal{A}(S_i)]}]$, *then*

$$\mathbb{E}_{x\sim\mathcal{D},\hat{x}_1,\hat{x}_2\in\mathcal{A}(x),\mathcal{C}(\hat{x}_1)\neq\mathcal{C}(\hat{x}_2)}[(\mu_{\mathcal{C}(\hat{x}_1)}^T \mu_{\mathcal{C}(\hat{x}_2)})^2] \leq 2\frac{\|\Lambda - I_d\|^2 + 2r^4\varepsilon^4 + K - d}{1 - \sum_i p_i^2},$$

This is a critical theorem of our study. As we know from Theorem 1, a large distance between classes ensures that samples of different classes are not misclassified. Theorem 2 gives a lower bound on the expectation of the inter-class distance, i.e. the upper bound on the cosine distance. We can reduce the expectation in Theorem 2 by optimizing the upper bound, so that the probability of the inter-class distance is large enough increases, which can further improve the generalization error rate.

By Theorem 2, if we can minimize $\|\Lambda - I_d\|$, the inter-class distances will be constrained. If the expectation is constrained, there cannot be a large number of inter-class distances that exceed this limit. Formally, we have the inequality for a non-negative random variable X:

$$\forall a > 0, \mathbb{P}[X > a] < \frac{\mathbb{E}[X]}{a}. \tag{10}$$

Combining the above two theorems, we can get the error rate theorem (See appendix for the proof) :

Theorem 3 *The error rate satisfies:*

$$Er(f) = \mathbb{P}[G(\hat{x}; f) \neq C(x)] \leq \frac{\frac{2}{d^2}\|\Lambda - I_d\|^2 + 2\varepsilon^4 + \frac{K-d}{d^2}}{(1 - \sum_i p_i^2)(1 - 4\varepsilon + \varepsilon^2)^2}.$$

Here as we know, ε is related to the strength of data augmentation. The remaining part of the upper bound shows that, given the dataset and data augmentation, the upper bound of the error rate is related to the following formula:

$$\mathcal{L}_{bound} = \frac{2}{d^2}\|\Lambda - I_d\|^2 + \frac{K - d}{d^2}. \tag{11}$$

Here we get that minimizing $\|\Lambda - I_d\|^2 = \mathcal{L}_{cor}$ can help reducing the error rate. As for d, if it is too small, that error rate bound will be larger than 1, thus the upper bound will become meaningless.

In the next section, we look into the loss function of contrastive learning models, and shows how they optimize $\mathcal{L}_{cor}$.

4.2 Loss Analysis

In this section we study the loss of Barlow Twins (Zbontar et al., 2021) and SimSiam (Chen & He, 2020), and explore how they guarantee their generalizability. Specifically, we will demonstrate how they optimize $\mathcal{L}_{cor}(f) = \|\Lambda - I_d\|^2 = \|\mathbb{E}_{x\sim\mathcal{D},\hat{x}_1\in\mathcal{A}(x)}[f(\hat{x}_1)f(\hat{x}_1)^T] - I_d\|^2$.

For Barlow Twins, the loss is

$$\mathcal{L}_{\mathcal{BT}} = \sum_{i\in[d]}(1-\mathcal{C}_{ii})^2 + \lambda \sum_{i\in[d]}\sum_{j\neq i}\mathcal{C}_{ij}^2. \tag{12}$$

Let $\lambda = 1$, and by definition of $\mathcal{C}$, it can be inferred that optimizing $\mathcal{L}_{\mathcal{BT}}$ leads to minimizing $\|\Lambda - I_d\|^2$ (See appendix for the proof) :

$$\mathcal{L}_{\mathcal{BT}} = \|\mathcal{C} - I_d\|^2 \geq \frac{1}{2}\|\Lambda - I_d\|^2 - \varepsilon^2. \tag{13}$$

Barlow Twins explicitly minimize the cross-correlation matrix loss, but SimSiam is more complicated:

$$\mathcal{L}_{SimSiam} = -\mathbb{E}_{x\sim\mathcal{D},\hat{x}_1,\hat{x}_2\in\mathcal{A}(x)}[\text{sim}(h(f(\hat{x}_1)), \text{sg}(f(\hat{x}_2)))]. \tag{14}$$

Let z_1, z_2 be the corresponding representations $f(\hat{x}_1), f(\hat{x}_2)$ of augmented views $\hat{x}_1, \hat{x}_2$ in Eqn. 14. We make the following assumption.

(1) A linear bias-free network serve as the projection layer. That is $p_1 = h(z_1) = W_p z_1$ in Eqn. 14 .

(2) Representation z is updated via backpropagation.

Let $g(z_1, z_2) = \frac{z^T W_p z_1}{\|Z_2\|\|W_p z_1\|} = \text{sim}(p_1, z_2)$. The loss function of SimSiam is

$$\mathcal{L}_{SimSiam} = -\mathbb{E}_{z_1,z_2}[g(z_1, sg(z_2))] \tag{15}$$

Because $Cor(z) = zz^T$ is symmetric, $\exists U \in R^{n\times n}, s.t. UU^T = U^T U = I_d$, U diagonalizes $Cor(z)$. That is

$$\Lambda = U^T Cor(z) U = diag\{\lambda_1, \lambda_2, ..., \lambda_n\} \tag{16}$$

The column of U, i.e.u_i, is an orthonormal basis of $R^{n\times n}$. $\alpha_z = U^T z$ is the coordinate of $z \in S^{n-1}$ under U. Then, we have the following:

$$\Lambda = U^T \mathbb{E}[U\alpha_z \alpha_z^T U^T]U = \mathbb{E}[\alpha_z \alpha_z^T] \tag{17}$$

Lemma 1 *If $A \in M_n(C([a,b]))$ is Hermitian with the property that the eigenvalues of $A(t)$ are distinct for all $t \in [a,b]$, then A is diagonalizable in $M_n(C([a,b]))$ (Grove & Pedersen, 1984)*

Here $M_n(C([a,b]))$ is the set of $n \times n$ matrices whose elements are continuous in $C[a,b]$. A diagonalizable A in $M_n(C([a,b]))$ means that there is a unitary element $U \in M_n(C[a,b])$, such that for each $t \in [a,b]$, $U(t)^* A(t) U(t)$ is diagonal.

According to Lemma 1, the correlation matrix $Cor(z) = zz^T$ is real-valued and symmetrical, and thus a Hermitian.

During training there exists continuous $U(t)$ that diagonalizes $Cor(z(t))$. Moreover, because $Cor(z(t))$ is a real-valued symmetric, $U(t)$ must be real-valued too. That is, we have a continuous basis $\{u_i(t)\}_{i=1}^n$ fir the training step t It also allow us to consider the local change of Λ by assuming that U is locally constant.

By denoting $\Lambda_p = U^T W_p U$, we have the following:

Theorem 4 *According to Lemma 1, the local dynamics of SimSiam is*

$$d\Lambda_p = r_1 \frac{\|z\|}{\|p\|}\mathbb{E}[\Delta\tilde{\alpha}_{z_1}^T] = r_1 \frac{\|z\|}{\|p\|}[(I - c\tilde{\Lambda}_p)\Lambda + \epsilon],$$

$$d\Lambda = r_2 \frac{1}{\|p\| \cdot \|z\|}(\Lambda_p^T \mathbb{E}[\Delta\tilde{\alpha}_{z_1}^T] + \mathbb{E}[\Delta\tilde{\alpha}_{z_1}^T]^T \Lambda_p),$$

where $\tilde{\Lambda}_p = \frac{\|z\|}{\|p\|}\Lambda_p$, $\Delta = \tilde{\alpha}_{z_2} - \tilde{\alpha}_{z_2}^T \tilde{\alpha}_{p_1} \cdot \tilde{\alpha}_{p_1}$, $\epsilon \triangleq \mathbb{E}[(\tilde{\alpha}_{z_2} - \tilde{\alpha}_{z_1})\tilde{\alpha}_{z_1}^T] \ll 1$,$c \triangleq {z_2}^T p_1$ is close to 1 and r_1, r_2 are learning rate of W_p, z_1, respectively.

Table 1: k-nearest neighbors accuracy on pretrained simplified SimSiam model and Barlow Twins model with different output dimension d. Dimensionality reduction has a more significant impact on datasets with more categories.

Dimension	Barlow Twins		Simsiam	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
2048	85.61	53.30	88.15	55.01
1024	84.60	52.54	88.18	54.50
512	82.79	48.75	87.63	54.74
256	81.05	45.22	85.99	54.12
128	78.94	40.67	85.78	53.84
64	77.19	36.25	85.05	52.91
32	72.73	33.34	84.56	51.63
16	67.94	29.62	83.20	48.14
8	62.49	25.17	82.14	44.45
4	57.16	21.13	76.05	28.30

Tian et al. (2021) shows that during training, the eigenspaces of W_p and Λ gradually aligns, that is, they can be diagonalized by the same matrix U. Thus we can assume $W_p = \text{diag}(\lambda_p^1, \ldots, \lambda_p^d)$ are diagonal, and then derive the element-wise form :

$$d\lambda_p = r_1 \frac{\|z\|}{\|p\|}[(1 - c\tilde{\lambda}_p^i)\lambda^i + \epsilon^i], \tag{18}$$

$$d\lambda = \frac{2r_2\lambda_p^i}{\|z\| \cdot \|p\|}[(1 - c\tilde{\lambda}_p^i)\lambda^i + \epsilon^i] = \frac{2r_2\lambda_p^i}{\|z\|^2}[(1 - c\tilde{\lambda}_p^i)\lambda^i + \epsilon^i], \tag{19}$$

There is a clear intuitive understanding of how uniformity is optimized with this local dynamics. If λ^i is close to each other, then the output representation distribution will be uniform. We begin by examining the simplest version. By definition, λ^i ¿ 0 always holds. Accordingly, there are three cases for Eqn. 18 and 19 ignoring ϵ^i.

(1) When $\tilde{\lambda}_p^i \leq 0$,$(1 - c\tilde{\lambda}_p^i)\lambda^i \geq 0$,$d\lambda^i \leq 0$ and $d\lambda_p^i \geq 0$.

(2) When $0 \leq \tilde{\lambda}_p^i \leq \frac{1}{c}$, $(1 - c\tilde{\lambda}_p^i)\lambda^i \geq 0$,$d\lambda^i \geq 0$ and $d\lambda_p^i \geq 0$.

(3) When $\tilde{\lambda}_p^i \geq \frac{1}{c}$,$(1 - c\tilde{\lambda}_p^i)\lambda^i \leq 0$,$d\lambda^i \leq 0$ and $d\lambda_p^i \leq 0$.

The equality holds only if $\tilde{\lambda}_p^i = 0$, $\tilde{\lambda}_p^i = \frac{1}{c}$ or $\lambda^i = 0$.

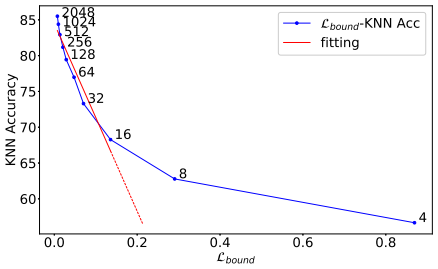
In conclusion, if $\tilde{\lambda}_p^i > \frac{1}{c}$ or $\tilde{\lambda}_p^i < 0$, λ^i deceases and λ_p^i moves to boundary 0 or $\frac{1}{c}$. If $\tilde{\lambda}_p^i \in (0, \frac{1}{c})$ or $\tilde{\lambda}_p^i < 0$, λ^i deceases and λ_p^i moves to boundary 0 or $\frac{1}{c}$, resulting in λ^i close to each other or $\lambda^i = 0$. That is, some dimensions collapse and others tend to have equivalent λ^i. So in those dimensions that do not collapse, the model optimize $\|\Lambda - I_d\|$.

Taking ϵ^i and weight decay into consideration, similar conclusion can be obtained. (See Appendix for the proof)

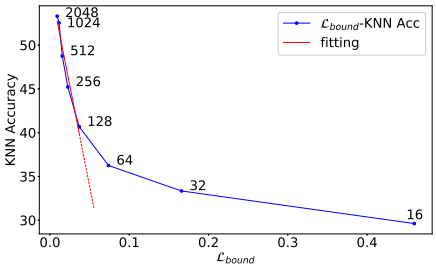
5 Experiments

In this section, we experimentally study the relationship between the upper bound of error rate and the actual generalization performance, indicating that the upper bound we obtained is a relatively tight bound, which has practical significance, and also explores the representation space dimension and the number of latent classes.

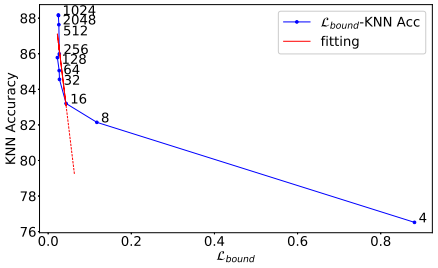
We conduct experiments on the CIFAR-10 and CIFAR-100 dataset (Krizhevsky et al., 2009) with classification as downstream tasks. We choose two independent augmentations of the same image as positive pairs, following the standard practice. We use the Barlow Twins model and simplified



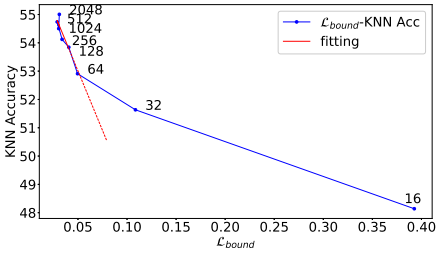
(a) Barlow Twins on CIFAR-10



(b) Barlow Twins on CIFAR-100



(c) Simsiam on CIFAR-10



(d) Simsiam on CIFAR-100

Figure 3: The relation between $\mathcal{L}_{bound}$ and knn accuracy of Barlow Twins and SimSiam model with different output dimensions. The blue dots are recorded $\mathcal{L}_{bound}$ and KNN accuracy with corresponding output dimension d aside, the solid red line is a linear function obtained by fitting the recorded data satisfying $d > K$ with the least squares method, and the dashed red line is the extension of the fitting linear function. Cases dimension = 4, 8 in the CIFAR-100 experiment are hidden to demonstrate the plot better.

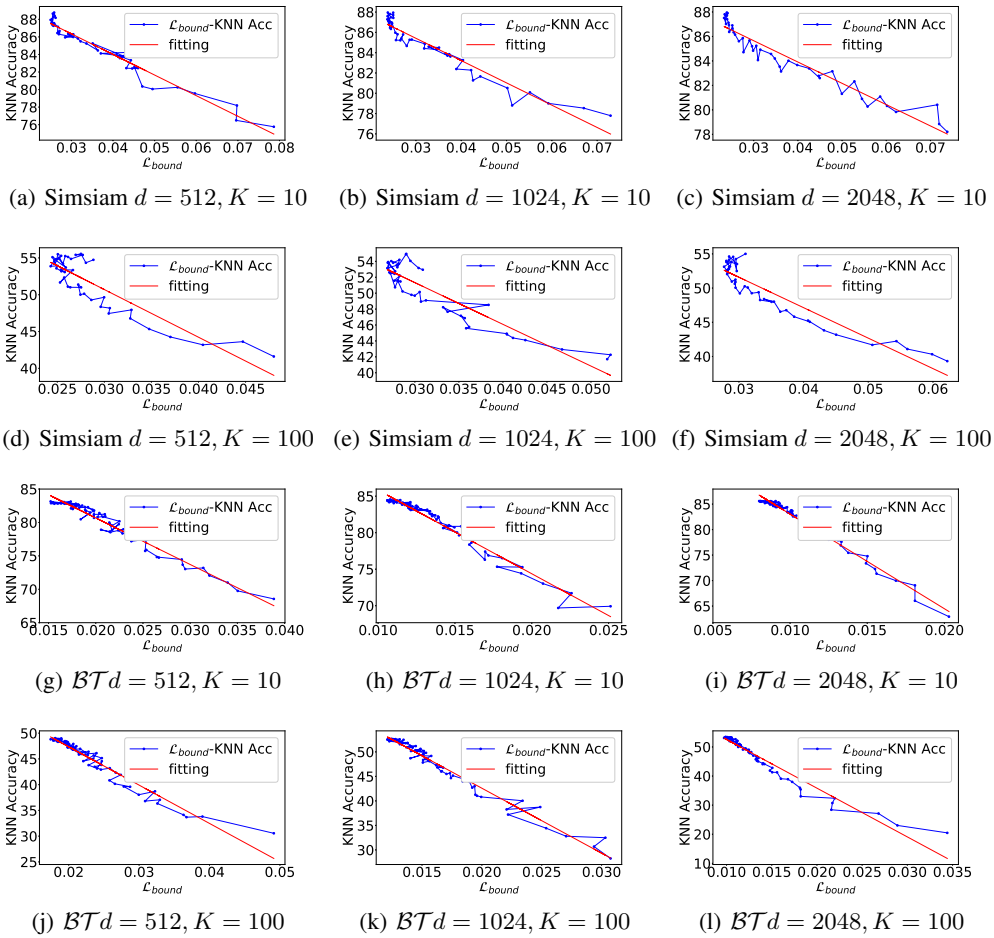
SimSiam model whose projector and predictor layer is set to be a linear network W_p to conduct the experiment. The simplified predictor layer maintains the optimization properties of the model while reducing the impact of additional factors such as batch normalization. We use ResNet-18 (He et al., 2016) as the encoder, run Barlow Twins model for 800 epochs and Simsiam models for 600 epochs and record the k-nearest neighbors(KNN) accuracy and $\mathcal{L}_{bound}$ in the minibatch. The code is modified on the basis of `https://github.com/facebookresearch/simsiam` and `https://github.com/IgorSusmelj/barlowtwins` . The results are recorded in Table 1. Note that in Theorem 3, except the dataset property p_i's, the dimension d and number of classes K plays a key role. If K is very large and d is relatively small, for example $K = 100$ and $d = 4$, then $\frac{K-d}{d^2} = 6$. the upper bound

$$\frac{\frac{2}{d^2}\|\Lambda - I_d\|^2 + 2\varepsilon^4 + \frac{K-d}{d^2}}{(1-\sum_i p_i^2)(1-4\varepsilon+\varepsilon^2)^2} > \frac{0+0+\frac{K-d}{d^2}}{1\times 1} = \frac{K-d}{d^2} \gg 1, \tag{20}$$

so that it could no longer give a meaningful bound of $Er(f) \in [0, 1]$.

Let $V(d) = \frac{K-d}{d^2}$ achieves its minimum value at $d = 2K$, $\lim_{d\to 0} V(d) = \infty$ and $\lim_{d\to\infty} V(d) = 0$. It induces that while $d > 2K$, the generalization performance does not hurt too much, but if d gets smaller, especially $d < K$, the model can no longer maintain its generalizability. In Table 1, CIFAR-10 and CIFAR-100 have 10 and 100 classes respectively, i.e. $K = 10,\ 100$ respectively. the minimum points are $d = 20, 200$. When $d > K$ for CIFAR-100, the KNN accuracy remains around 43%. When d gets lower, the accuracy drops quickly. As for CIFAR-10, a $32 - dim$ model performs as well as hundreds of dimensions. The accuracy drops when $d < 2K = 20$, this matches our prediction.

In Fig. 3 and Fig. 4, we did scatter plots of knn accuracy and $\mathcal{L}_{bound}$ on CIFAR-10 and CIFAR-100 datasets. The points in Fig. 3 correspond to the models in Table 1. Although it is not strict, there is still a clear correlation between knn accuracy and $\mathcal{L}_{bound}$ in the figure. The smaller $\mathcal{L}_{bound}$ is,



(a) Simsiam $d = 512, K = 10$ (b) Simsiam $d = 1024, K = 10$ (c) Simsiam $d = 2048, K = 10$

(d) Simsiam $d = 512, K = 100$ (e) Simsiam $d = 1024, K = 100$ (f) Simsiam $d = 2048, K = 100$

(g) $\mathcal{BT}d = 512, K = 10$ (h) $\mathcal{BT}d = 1024, K = 10$ (i) $\mathcal{BT}d = 2048, K = 10$

(j) $\mathcal{BT}d = 512, K = 100$ (k) $\mathcal{BT}d = 1024, K = 100$ (l) $\mathcal{BT}d = 2048, K = 100$

Figure 4: The relation between $\mathcal{L}_{bound}$ and knn accuracy of Barlow Twins and SimSiam models with different output dimensions in training epochs. The blue dots are $\mathcal{L}_{bound}$ and KNN accuracy recorded every 10 epoch in the training process and connected in chronological order, the red line is a linear function obtained by fitting the recorded data after 50 epochs for Barlow Twins and 150 epochs for Simsiam with the least squares method, due to uniformity metric at the initial stage of training not conforming to the assumption that positive sample pairs are close to each other.

the larger knn accuracy is. Note that by Theorem 3, $\mathcal{L}_{bound}$ upper bounds the generalization error rate. If $\mathcal{L}_{bound}$ becomes smaller, then the error rate will also become smaller, which means that the accuracy increases. If this upper bound is too loose, its value will not be so significantly related to the knn accuracy. So the theorem agrees with the experiment, the relationship between the upper bound and the knn accuracy is significant.

In Fig. 4, the linear relationship among the sample points is significant, and the closer the sample points are to the end of training (upper left), the closer they are to the fitted line. This shows that Theorem 3 can accurately estimate the generalization error, and the upper bound of the error rate we proposed can determine the actual error rate to some extent. On the other hand, as training progresses proceeding, sample points determined by $\mathcal{L}_{bound}$ and KNN accuracy move from right to left along the fitting line, $\mathcal{L}_{bound}$ decreases and K-nearest neighbor accuracy increases (corresponding to a decrease in error rate). The optimization process described in Section 4.2 for $\mathcal{L}_{bound}$ is validated.

6 CONCLUSION

In this paper we study the generalization of non-contrastive learning methods represented by Barlow Twins and SimSiam. We give an upper bound on the generalization error rate to illustrate the effect of inter-class distance on generalization performance. We demonstrate the relationship between the cross-correlation matrix of output features and inter-class distance. Then we analyze how non-contrastive learning methods optimize correlation so that they are sufficient to produce good generalization. Moreover, we find that the dimension of feature space and the number of latent classes also affect the generalizability. We conduct an experiment on CIFAR-10 and CIFAR-100 dataset to verify our theoretical results.

REFERENCES

Sanjeev Arora, Hrishikesh Khandeparkar, Mikhail Khodak, Orestis Plevrakis, and Nikunj Saunshi. A theoretical analysis of contrastive unsupervised representation learning. *arXiv preprint arXiv:1902.09229*, 2019.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020.

Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. *arXiv preprint arXiv:2011.10566*, 2020.

Zewen Chi, Li Dong, Furu Wei, Nan Yang, Saksham Singhal, Wenhui Wang, Xia Song, Xian-Ling Mao, Heyan Huang, and Ming Zhou. Infoxlm: An information-theoretic framework for cross-lingual language model pre-training. *arXiv preprint arXiv:2007.07834*, 2020.

Ching-Yao Chuang, Joshua Robinson, Yen-Chen Lin, Antonio Torralba, and Stefanie Jegelka. Debiased contrastive learning. *Advances in neural information processing systems*, 33:8765–8775, 2020.

Henry Cohn and Abhinav Kumar. Universally optimal distribution of points on spheres. *Journal of the American Mathematical Society*, 20(1):99–148, 2007.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*, 2020.

Karsten Grove and Gert Kjærgård Pedersen. Diagonalizing matrices over c(x). *Journal of Functional Analysis*, 59(1):65–89, 1984. ISSN 0022-1236.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9729–9738, 2020.

Qianjiang Hu, Xiao Wang, Wei Hu, and Guo-Jun Qi. Adco: Adversarial contrast for efficient learning of unsupervised representations from self-trained negative adversaries. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1074–1083, 2021.

Weiran Huang, Mingyang Yi, and Xuyang Zhao. Towards the generalization of contrastive self-supervised learning. *arXiv preprint arXiv:2111.00743*, 2021.

Tri Huynh, Simon Kornblith, Matthew R Walter, Michael Maire, and Maryam Khademi. Boosting contrastive self-supervised learning with false negative cancellation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2785–2795, 2022.

Dan Iter, Kelvin Guu, Larry Lansing, and Dan Jurafsky. Pretraining with contrastive sentence objectives improves discourse performance of language models. *arXiv preprint arXiv:2005.10389*, 2020.

Li Jing, Pascal Vincent, Yann LeCun, and Yuandong Tian. Understanding dimensional collapse in contrastive self-supervised learning. *arXiv preprint arXiv:2110.09348*, 2021.

Yannis Kalantidis, Mert Bulent Sariyildiz, Noe Pion, Philippe Weinzaepfel, and Diane Larlus. Hard negative mixing for contrastive learning. *Advances in Neural Information Processing Systems*, 33:21798–21809, 2020.

Seongjae Kim, Jinseok Seol, Holim Lim, and Sang-goo Lee. False negative distillation and contrastive learning for personalized outfit recommendation. *arXiv preprint arXiv:2110.06483*, 2021.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Aaqib Saeed, David Grangier, and Neil Zeghidour. Contrastive learning of general-purpose audio representations. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3875–3879. IEEE, 2021.

Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *European conference on computer vision*, pp. 776–794. Springer, 2020a.

Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning? *Advances in Neural Information Processing Systems*, 33:6827–6839, 2020b.

Yuandong Tian, Xinlei Chen, and Surya Ganguli. Understanding self-supervised learning dynamics without contrastive pairs. *arXiv preprint arXiv:2102.06810*, 2021.

Christopher Tosh, Akshay Krishnamurthy, and Daniel Hsu. Contrastive learning, multi-view redundancy, and linear models. In *Algorithmic Learning Theory*, pp. 1179–1206. PMLR, 2021.

Aaron Van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv e-prints*, pp. arXiv–1807, 2018.

Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, pp. 9929–9939. PMLR, 2020.

Tete Xiao, Xiaolong Wang, Alexei A Efros, and Trevor Darrell. What should not be contrastive in contrastive learning. *arXiv preprint arXiv:2008.05659*, 2020.

Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. Graph contrastive learning with augmentations. *Advances in Neural Information Processing Systems*, 33:5812–5823, 2020.

Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. *arXiv preprint arXiv:2103.03230*, 2021.

Yanqiao Zhu, Yichen Xu, Feng Yu, Qiang Liu, Shu Wu, and Liang Wang. Deep graph contrastive representation learning. *arXiv preprint arXiv:2006.04131*, 2020.

7 APPENDIX

A PROOF OF THEOREM 1

Proof 1 *Without loss of generality, we only prove that if* $\mu_i^T \mu_j < r^2(1-4\varepsilon+\varepsilon^2),\ S_i \sim \mathcal{C}_i,\ \forall \hat{x} \in \mathcal{A}(S_i)$, $G(\hat{x}; f) \neq j$.

$\forall k \in [K]$,

$$\begin{aligned} \|\mu_k\| &= \|\mathbb{E}_{\hat{x}\in\mathcal{A}(S_k)}[f(x)]\| \\ &= \|f(x_0) + \mathbb{E}_{\hat{x}\in\mathcal{A}(S_k)}[f(x) - f(x_0)]\| \\ &\geq \|f(x_0)\| - r\varepsilon \\ &= r(1-\varepsilon) \end{aligned} \tag{21}$$

$$\begin{aligned} &\|f(\hat{x}) - \mu_i\|^2 - \|f(\hat{x}) - \mu_j\|^2 \\ =&2f(\hat{x})^T\mu_j - 2f(\hat{x})^T\mu_i + \|\mu_i\|^2 - \|\mu_j\|^2 \\ =&2(f(\hat{x})^T - \mu_i^T + \mu_i^T)(\mu_j - \mu_i) + \|\mu_i\|^2 - \|\mu_j\|^2 \\ =&2(f(\hat{x})^T - \mu_i^T)(\mu_j - \mu_i) + 2\mu_i^T\mu_j - \|\mu_i\|^2 - \|\mu_j\|^2 \\ \leq&2r\varepsilon\|\mu_j - \mu_i\| + 2\mu_i^T\mu_j - 2r^2(1-\varepsilon)^2 \\ \leq&2\mu_i^T\mu_j + 4r^2\varepsilon - 2r^2(1-\varepsilon)^2 \\ \leq&2\mu_i^T\mu_j + 4r^2\varepsilon - 2r^2(1-\varepsilon)^2 \\ <&0 \end{aligned} \tag{22}$$

That is, $\|f(\hat{x}) - \mu_i\|^2 < \|f(\hat{x}) - \mu_j\|^2$*, so we have for* $\hat{x} \in \mathcal{A}(S_i)$, $G(\hat{x}; f) \neq j$

B PROOF OF THEOREM 2

Proof 2 *First we have:*

$$\mathbb{E}_{x\sim\mathcal{D},\hat{x}_1,\hat{x}_2\in\mathcal{A}(x),\mathcal{C}(\hat{x}_1)\neq\mathcal{C}(\hat{x}_2)}[(\mu_{\mathcal{C}(\hat{x}_1)}^T\mu_{\mathcal{C}(\hat{x}_2)})^2] = \frac{1}{1-\sum_i p_i^2}\sum_{i=1}^{K}\sum_{j\neq i} p_i p_j (\mu_i\mu_j)^2 \tag{23}$$

As for $\|\Lambda - I_d\|^2$*:*

$$\begin{aligned} \|\Lambda - I_d\|^2 &= \|\mathbb{E}_{x\sim\mathcal{D},\hat{x}_1\in\mathcal{A}(x)}[f(\hat{x})f(\hat{x})^T] - I_d\|^2 \\ &= \|\sum_{i\in[K]} p_i \mathbb{E}_{x\sim\mathcal{C}_i,\hat{x}_1\in\mathcal{A}(x)}[f(\hat{x})f(\hat{x})^T] - I_d\|^2 \\ &= \|\sum_{i\in[K]} p_i \mathbb{E}_{x\sim\mathcal{C}_i,\hat{x}_1\in\mathcal{A}(x)}[(\mu_i + f(\hat{x}) - \mu_i)(\mu_i + f(\hat{x}) - \mu_i)^T] - I_d\|^2 \\ &= \|\sum_{i\in[K]} p_i\mu_i\mu_i^T + \sum_{i\in[K]} p_i \mathbb{E}_{x\sim\mathcal{C}_i,\hat{x}_1\in\mathcal{A}(x)}(f(\hat{x}) - \mu_i)(f(\hat{x}) - \mu_i)^T - I_d\|^2 \\ &\geq \frac{1}{2}\|\sum_{i\in[K]} p_i\mu_i\mu_i^T - I_d\|^2 - r^4\varepsilon^4 \end{aligned} \tag{24}$$

The last inequality is because for any A, B*,*

$$\|A + B\|^2 \geq \frac{1}{2}\|A\|^2 - \|B\|^2 \tag{25}$$

Let $A = (\sqrt{p_1}\mu_1, \ldots, \sqrt{p_K}\mu_K)$*, we have*

$$\begin{aligned} \| \sum_{i\in[K]} p_i\mu_i\mu_i^T - I_d\|^2 &= \|AA^T - I_d\|^2 \\ &= tr((AA^T - I_d)(AA^T - I_d)^T) \\ &= tr(AA^TAA^T - 2AA^T + I_d) \\ &= tr(A^TAA^TA - 2A^TA + I_K) - tr(I_K) + tr(I_d) \\ &= \|A^TA - I_K\|^2 - K + d \\ &\geq \sum_{i\in[K]}\sum_{j\neq i} p_ip_j(\mu_i^T\mu_j)^2 - K + d \end{aligned} \tag{26}$$

Finally, it is obvious that:

$$\begin{aligned} \sum_{i\in[K]}\sum_{j\neq i} p_ip_j(\mu_i^T\mu_j)^2 &\leq \| \sum_{i\in[K]} p_i\mu_i\mu_i^T - I_d\|^2 + K - d \\ &\leq 2\|\Lambda - I_d\|^2 + 2r^4\varepsilon^4 + K - d \end{aligned} \tag{27}$$

C Proof of Theorem 3

Proof 3

$$\begin{aligned} Er(f) &= \mathbb{P}[G(x;f) \neq C(x)] \\ &= \mathbb{P}[G(\hat{x}) = j, \mathcal{C}(\hat{x}) = i \neq j] \\ &\leq \mathbb{P}[(\mu_i^T\mu_j)^2 \geq r^4(1 - 4\varepsilon + \varepsilon^2)^2] \\ &\leq \frac{\mathbb{E}[(\mu_i^T\mu_j)^2]}{r^4(1 - 4\varepsilon + \varepsilon^2)^2} \\ &\leq \frac{\frac{2}{d^2}\|\Lambda - I_d\|^2 + 2\varepsilon^4 + \frac{K-d}{d^2}}{(1 - \sum_i p_i^2)(1 - 4\varepsilon + \varepsilon^2)^2} \end{aligned} \tag{28}$$

D Proof of Equation 13

Proof 4

$$\begin{aligned} \mathcal{L}_{\mathcal{BT}} &= \|\mathcal{C} - I_d\|^2 \\ &= \|\mathbb{E}_{x\sim\mathcal{D},\hat{x}_1,\hat{x}_2\in\mathcal{A}(x)}[f(\hat{x}_1)f(\hat{x}_2)^T] - I_d\|^2 \\ &= \|\mathbb{E}_{x\sim\mathcal{D},\hat{x}_1,\hat{x}_2\in\mathcal{A}(x)}[f(\hat{x}_1)f(\hat{x}_1)^T + f(\hat{x}_1)(f(\hat{x}_2) - f(\hat{x}_1))^T] - I_d\|^2 \\ &\geq \frac{1}{2}\|\mathbb{E}_{x\sim\mathcal{D},\hat{x}\in\mathcal{A}(x)}[f(\hat{x})f(\hat{x})^T] - I_d\|^2 - \|\mathbb{E}_{x\sim\mathcal{D},\hat{x}_1,\hat{x}_2\in\mathcal{A}(x)}[f(\hat{x}_1)(f(\hat{x}_2) - f(\hat{x}_1))^T]\|^2 \\ &\geq \frac{1}{2}\|\mathbb{E}_{x\sim\mathcal{D},\hat{x}\in\mathcal{A}(x)}[f(\hat{x})f(\hat{x})^T] - I_d\|^2 - \varepsilon^2 \\ &= \frac{1}{2}\|\Lambda - I_d\|^2 - \varepsilon^2 \end{aligned} \tag{29}$$

E Local dynamics analysis of Simsiam

E.1 Prove of local dynamics analysis of Simsiam

Proof 5 *First we note three basic formula:*

(1) Derivation of the unit vector: $\forall x \in R^n,\ d\tilde{x} = \frac{dx - \tilde{x}^T dx \cdot \tilde{x}}{\|x\|} = \frac{I - \tilde{x}\tilde{x}^T}{\|x\|} \Rightarrow$

$$\frac{\partial \tilde{x}}{\partial x} = \frac{I - \tilde{x}\tilde{x}^T}{\|x\|} \tag{30}$$

(2) Chain rule: $\forall$*vector* $x, y, f(y(x)), df = (\frac{\partial f}{\partial y})^T dy = (\frac{\partial f}{\partial y})^T (\frac{\partial y}{\partial x})^T dx \Rightarrow$

$$\frac{\partial f}{\partial x} = \frac{\partial f}{\partial y}\frac{\partial y}{\partial x} \tag{31}$$

(3) AXB type: $\forall f(Y) \in R, Y = AXB, df = tr((\frac{\partial f}{\partial Y})^T dY) = tr((\frac{\partial f}{\partial Y})^T AdXB) = tr((B\frac{\partial f}{\partial Y})^T AdX) \Rightarrow$

$$\frac{\partial f}{\partial X} = A^T \frac{\partial f}{\partial Y} B^T \tag{32}$$

Through the formulas above we have

$$\begin{aligned} \frac{\partial g}{\partial p_1} &= \frac{\partial g}{\partial \tilde{p}_1}\frac{\partial \tilde{p}_1}{\partial p_1} \\ &= \frac{I - \tilde{p}_1\tilde{p}_1^T}{\|p_1\|}\tilde{z}_2 \\ &= \frac{1}{\|p_1\|}(\tilde{z}_2 - \tilde{z}_2^T\tilde{p}_1 \cdot \tilde{p}_1) \end{aligned} \tag{33}$$

Because $p_1 = W_p z_1$*, We have* $\frac{\partial g}{\partial z_1} = W_p^T \frac{\partial g}{\partial p_1}$ *and* $\frac{\partial g}{\partial W_p} = \frac{\partial g}{\partial p_i} z_1^T$*, thus*

$$\frac{\partial g}{\partial z_1} = \frac{\|z_1\|}{\|p_1\|}(\tilde{z}_2 - \tilde{z}_2^T\tilde{p}_1 \cdot \tilde{p}_1)\tilde{z}_1^T \tag{34}$$

$$\frac{\partial g}{\partial W_p} = \frac{\|W_p^T\|}{\|p_1\|}(\tilde{z}_2 - \tilde{z}_2^T\tilde{p}_1 \cdot \tilde{p}_1) \tag{35}$$

Λ_p *and* α_{z_i} *update via:*

$$\begin{aligned} d\Lambda_p &= r_1 \mathbb{E}[U^T \frac{\partial g}{\partial W_p} U] \\ &= r_1 \mathbb{E}[\frac{\|\alpha_{z_1}\|}{\|\alpha_{p_1}\|}(\tilde{\alpha}_{z_2} - \tilde{\alpha}_{z_2}^T\tilde{\alpha}_{p_1} \cdot \tilde{\alpha}_{p_1})\tilde{\alpha}_{z_1}^T] \\ &= r_1 \mathbb{E}[\frac{\|\alpha_{z_1}\|}{\|\alpha_{p_1}\|}\Delta\tilde{\alpha}_{z_1}^T] \end{aligned} \tag{36}$$

$$\begin{aligned} d\alpha_{z_1} &= r_2 U^T \frac{\partial g}{\partial z_1} \\ &= r_2 \frac{\Lambda_p^T}{\|\alpha_{p_1}\|}(\tilde{\alpha}_{z_2} - \tilde{\alpha}_{z_2}^T\tilde{\alpha}_{p_1} \cdot \tilde{\alpha}_{p_1}) \\ &= r_2 \frac{\Lambda_p^T}{\|\alpha_{p_1}\|}\Delta \end{aligned} \tag{37}$$

where $\Delta = \tilde{\alpha}_{z_2} - \tilde{\alpha}_{z_2}^T\tilde{\alpha}_{p_1} \cdot \tilde{\alpha}_{p_1}$ *and* r_1*,* r_2 *are learning rate of* W_p*,* z_1*, respectively. As for* Λ*, we need the dynamic of* $\tilde{\alpha}_{z_1}$*. For any vector* x*,* $\frac{d\|x\|}{dx} = \frac{x}{\|x\|}$*,* $\frac{d\frac{x}{\|x\|}}{dx} = \frac{\|x\|^2 I - xx^T}{\|x\|^3}$*. So*

$$d\tilde{\alpha}_{z_1} = (\frac{d\tilde{\alpha}_{z_1}}{d\alpha_{z_1}})^T d\alpha_{z_1} = r_2(I - \tilde{\alpha}_{z_1}\tilde{\alpha}_{z_1}^T)\frac{\Lambda_p^T}{\|\alpha_{z_1}\|\|\alpha_{p_1}\|}\Delta \tag{38}$$

$$d\Lambda = \mathbb{E}[d(\tilde{\alpha}_{z_1})\tilde{\alpha}_{z_1}^T] + \mathbb{E}[\tilde{\alpha}_{z_1} d(\tilde{\alpha}_{z_1}^T)] \tag{39}$$

$$\begin{aligned}\mathbb{E}[d(\tilde{\alpha}_{z_1})\tilde{\alpha}_{z_1}^T] &= r_2\mathbb{E}[(I-\tilde{\alpha}_{z_1}\tilde{\alpha}_{z_1}^T)\frac{\Lambda_p^T}{\|\alpha_{z_1}\|\|\alpha_{p_1}\|}\Delta\tilde{\alpha}_{z_1}^T] \\ &= r_2\mathbb{E}[\frac{\Lambda_p^T}{\|\alpha_{z_1}\|\|\alpha_{p_1}\|}\Delta\tilde{\alpha}_{z_1}^T]\end{aligned} \tag{40}$$

The last equality in Eqn. 40 is due to:

$$\begin{aligned}\tilde{\alpha}_{z_1}\tilde{\alpha}_{z_1}^T\frac{\Lambda_p^T}{\|\alpha_{z_1}\|^2\|\alpha_{p_1}\|}\Delta\tilde{\alpha}_{z_1}^T &= \tilde{\alpha}_{z_1}\frac{\alpha_{z_1}\Lambda_p^T}{\|\alpha_{z_1}\|\|\alpha_{p_1}\|}\Delta\tilde{\alpha}_{z_1}^T \\ &= \frac{\tilde{\alpha}_{z_1}}{\|\tilde{\alpha}_{z_1}\|^2}\tilde{\alpha}_{p_1}^T(\tilde{\alpha}_{z_2}-\tilde{\alpha}_{z_2}^T\tilde{\alpha}_{p_1}\cdot\tilde{\alpha}_{p_1})\tilde{\alpha}_{z_1}^T \\ &= \frac{\tilde{\alpha}_{z_1}}{\|\tilde{\alpha}_{z_1}\|^2}\tilde{\alpha}_{p_1}^T\tilde{\alpha}_{z_2}-\tilde{\alpha}_{z_2}^T\tilde{\alpha}_{p_1}\cdot\tilde{\alpha}_{p_1}^T\tilde{\alpha}_{p_1}\tilde{\alpha}_{z_1}^T \\ &= 0\end{aligned} \tag{41}$$

The function $\Delta\tilde{\alpha}_{z_1}^T = (\tilde{\alpha}_{z_2}-\tilde{\alpha}_{z_2}^T\tilde{\alpha}_{p_1}\cdot\tilde{\alpha}_{p_1})\tilde{\alpha}_{z_1}^T$ is formally independent of $\|z_1\|$ and $\|p_1\|$. So we assume the distribution of $\Delta\tilde{\alpha}_{z_1}^T$ is independent of $\|z_1\|$ and $\|p_1\|$. So we have

$$d\Lambda_p = r_1\mathbb{E}_{\|z_1\|,\|p_1\|}[\frac{\|z_1\|}{\|p_1\|}]\mathbb{E}[\Delta\tilde{\alpha}_{z_1}^T] \tag{42}$$

$$\mathbb{E}[d(\tilde{\alpha}_{z_1})\tilde{\alpha}_{z_1}^T] = r_2\mathbb{E}_{\|z_1\|,\|p_1\|}[\frac{\Lambda_p^T}{\|z_1\|\|p_1\|}]\mathbb{E}[\Delta\tilde{\alpha}_{z_1}^T] \tag{43}$$

For simplicity, we ignore $\mathbb{E}_{\|z_1\|,\|p_1\|}[\cdot]$ and denote $\|z_1\| = \|z_\|$, $\|p_1\| = \|p_\|$ in the following. It seems to be a strong assumption, but this can be regarded to analyze the dynamics of the part of points whose $\|z_\|$ and $\|p_\|$ are certain values. Under previous assumption that $\Delta\tilde{\alpha}_{z_1}^T$ is independent of $\|z_1\|$ and $\|p_1\|$, it is a natural simplification. Eqn. 36 and 40 become:

$$d\Lambda_p = r_1\mathbb{E}[\frac{\|z_1\|}{\|p_1\|}\Delta\tilde{\alpha}_{z_1}^T] \tag{44}$$

$$\mathbb{E}[d(\tilde{\alpha}_{z_1})\tilde{\alpha}_{z_1}^T] = r_2\mathbb{E}[\frac{\Lambda_p^T}{\|z_1\|\|p_1\|}\Delta\tilde{\alpha}_{z_1}^T] \tag{45}$$

Both of them have the term $\mathbb{E}[\Delta\tilde{\alpha}_{z_1}^T]$, which is crucial for the dynamics. We examine it closely:

$$\mathbb{E}[\Delta\tilde{\alpha}_{z_1}^T] = \mathbb{E}[\tilde{\alpha}_{z_2}\tilde{\alpha}_{z_1}^T-\tilde{z}_2^T\tilde{p}_1\cdot\tilde{\alpha}_{p_1}\tilde{\alpha}_{z_1}^T] = \mathbb{E}[\tilde{\alpha}_{z_2}\tilde{\alpha}_{z_1}^t]-\mathbb{E}[\tilde{z}_2^T\tilde{p}_1\cdot\tilde{\alpha}_{p_1}\tilde{\alpha}_{z_1}^T] \tag{46}$$

In the training phase, we pull $\tilde{z}_2$ and $\tilde{p}_1$ together. $\tilde{z}_2\tilde{p}_1$ approaches 1 at the very beginning and remains close to 1 during the whole training phase. So we assume that $z_2^Tp_1 = c$ is close to 1. For the term $\mathbb{E}[\tilde{\alpha}_{z_2}\tilde{\alpha}_{z_1}^T]$, z_1, z_2 are the representations of different views of the same image through the same model, We assume that $\tilde{\alpha}_{z_2} = \tilde{\alpha}_{z_1}+\varepsilon$ and $\|\varepsilon\| \ll 1$ is related to the intensity of the data augmentation. Moreover, $\tilde{\alpha}_{p_1} = \frac{\Lambda_p\alpha_{z_1}}{\|p\|} = \frac{\|z\|}{\|p\|}\Lambda_p\tilde{\alpha}_{z_1}$. Denoting $\tilde{\Lambda}_p \triangleq \frac{\|z\|}{\|p\|}\Lambda_p$, $\epsilon \triangleq \mathbb{E}[\varepsilon\tilde{\alpha}_{z_1}^T]$, we have

$$\mathbb{E}[\tilde{\alpha}_{z_1}^T] = \Lambda - c\tilde{\Lambda_p}\Lambda + \mathbb{E}[\varepsilon\tilde{\alpha}_{z_1}] = (I-c\tilde{\Lambda}_p)\Lambda+\epsilon \tag{47}$$

Substituting Eqn. 47 to Eqn. 19 and 18, we finally get the local dynamics of Λ_p and Λ

E.2 Taking ϵ^i and weight decay into consideration

In the preceding section, we present an intuitive description of how Λ and Λ_p change, without addressing ϵ. SimSiam use weight decay during training, which has an impact on the models' performance. In this section, we take them into consideration and come up with similar results. We can obtain the similar local dynamics when there is weight decay η.

$$\begin{aligned}d\lambda_p &= r_1\frac{\|z\|}{\|p\|}[(1-c\tilde{\lambda}_p^i)\lambda^i+\epsilon^i]-r_1\eta\frac{\|p\|}{\|z\|}\tilde{\lambda}_p^i \\ &= r_1\frac{\|z\|}{\|p\|}[(1-c\tilde{\lambda}_p^i)\lambda^i+\epsilon^i-\eta\frac{\|p\|^2}{\|z\|^2}\tilde{\lambda}_p^i],\end{aligned} \tag{48}$$

$$
\begin{aligned}
d\lambda &= r_2 \frac{\lambda_p^i}{\|z\| \cdot \|p\|}[(1 - c\tilde{\lambda}_p^i)\lambda^i + \epsilon^i] - 2r_2\eta\lambda^i \\
&= 2r_2\lambda^i[\frac{\lambda_p^i}{\|z\|^2}[(1 - c\tilde{\lambda}_p^i) + \frac{\epsilon^i}{\lambda^i}] - \eta],
\end{aligned} \tag{49}
$$

Proof 6 *With weight decay η, similar to the previous section, we have*

$$
d\Lambda_p = r_1\mathbb{E}[\frac{\|\alpha_{z_1}\|}{\|\alpha_{p_1}\|}\Delta\tilde{\alpha}_{z_1}^T] - r_1\eta\Lambda_p \tag{50}
$$

$$
d\alpha_{z_1} = r_2\frac{\Lambda_p^T}{\|\alpha_{p_1}\|}\Delta - r_2\eta\alpha_{z_1} \tag{51}
$$

Note that

$$
\begin{aligned}
(I - \tilde{\alpha}_{z_1}\tilde{\alpha}_{z_1}^T)\tilde{\alpha}_{z_1}\tilde{\alpha}_{z_1}^T &= \tilde{\alpha}_{z_1}\tilde{\alpha}_{z_1}^T - \tilde{\alpha}_{z_1}(\tilde{\alpha}_{z_1}^T\tilde{\alpha}_{z_1})\tilde{\alpha}_{z_1}^T \\
&= \tilde{\alpha}_{z_1}\tilde{\alpha}_{z_1}^T - \tilde{\alpha}_{z_1}\tilde{\alpha}_{z_1}^T = 0
\end{aligned} \tag{52}
$$

Then, by Eqn. 41, $\tilde{\alpha}_{z_1}\tilde{\alpha}_{z_1}^T\frac{\Lambda_p^T}{\|\alpha_{z_1}\|^2\|\alpha_{p_1}\|}\Delta\tilde{\alpha}_{z_1}^T$, so we have

$$
\begin{aligned}
\mathbb{E}[d(\tilde{\alpha}_{z_1})\tilde{\alpha}_{z_1}^T] &= \mathbb{E}[\frac{\partial\tilde{\alpha}_{z_1}}{\partial\alpha_{z_1}}d\alpha_{z_1}\tilde{\alpha}_{z_1}^T] \\
&= r_2\mathbb{E}[\frac{I - \tilde{\alpha}_{z_1}\tilde{\alpha}_{z_1}^T}{\|\alpha_{z_1}\|}(\frac{\Lambda_p^T}{\|\alpha_{z_1}\|\|\alpha_{p_1}\|}\Delta\tilde{\alpha}_{z_1}^T - \eta\tilde{\alpha}_{z_1}\tilde{\alpha}_{z_1}^T)]
\end{aligned} \tag{53}
$$

Similar to the previous section, the local dynamics can be obtained:

$$
d\Lambda_p = r_1\frac{\|z\|}{\|p\|}[(I - c\tilde{\Lambda}_p)\Lambda + \epsilon] - r_1\eta\Lambda_p \tag{54}
$$

$$
d\Lambda = \frac{r_2}{\|z\|\|p\|}(\Lambda_p^T[(I - c\tilde{\Lambda}_p)\Lambda + \epsilon] + [(I - c\tilde{\Lambda}_p)\Lambda + \epsilon]^T\Lambda_p) - 2r_2\eta\Lambda \tag{55}
$$

Similarly assuming $\Lambda_p = diag(\lambda_p^1, ..., \lambda_p^n)$ as in Sec 4.2. Eqn. 48 and 49 can be obtained.

Note that $\epsilon = \mathbb{E}[\varepsilon\tilde{\alpha}_{z_1}^T]$, $\varepsilon = \tilde{\alpha}_{z_2} - \tilde{\alpha}_{z_1}$ is generated by the output of the model on the disturbed samples, and randomness operates on the samples rather than directly on $\tilde{\alpha}_{z_1}$. However, the significance of each component varies depending on the model. Thus, we may assume that ϵ^i will be smaller than λ^i specifically $\exists e > 0$, $|\frac{\epsilon^i}{\lambda^i}| < e \ll 1$. New terms affect the boundaries of different phases, but the intuitive understanding remains unchanged. We have

$$
d\lambda_p^i > 0 \Leftrightarrow (1 - c\tilde{\lambda}_p^i + \frac{\epsilon^i}{\lambda^i})\lambda^i > \eta\frac{\|p\|^2}{\|z\|^2}\tilde{\lambda}_p^i \tag{56}
$$

$$
d\lambda^i > 0 \Leftrightarrow \frac{\tilde{\lambda}_p^i}{\|z\|^2}(1 - c\tilde{\lambda}_p^i + \frac{\epsilon^i}{\lambda^i}) > \eta \tag{57}
$$

Similarly, there are three cases for Eqn. 48 and 49:

(1) When $\tilde{\lambda}_p^i \leq 0$, Eqn. 56 holds and Eqn. 57 fails. $d\lambda^i \leq 0$ and $d\lambda_p^i \geq 0$.

(2) When $0 \leq \tilde{\lambda}_p^i \leq \frac{1}{c}$, $d\lambda_p^i > 0 \Leftrightarrow \lambda^i > \eta\frac{\|p\|^2}{\|z\|^2}/(\frac{1+\frac{\epsilon^i}{\lambda^i}}{\tilde{\lambda}_p^i} - c)$, which more likely holds when $\tilde{\lambda}_p^i$ is near 0. $d\lambda^i > 0$ more likely holds when $\tilde{\lambda}_p^i$ is not too close to the boundary 0 and $\frac{1+\frac{\epsilon^i}{\lambda^i}}{c}$.

(3) When $\tilde{\lambda}_p^i \geq \frac{1}{c}$, both Eqn. 56 and 57 fail. $d\lambda^i \leq 0$ and $d\lambda_p^i \leq 0$.

E.3 VISUALIZATION OF THE CORRELATION MATRIX OF SIMSIAM REPRESENTATIONS

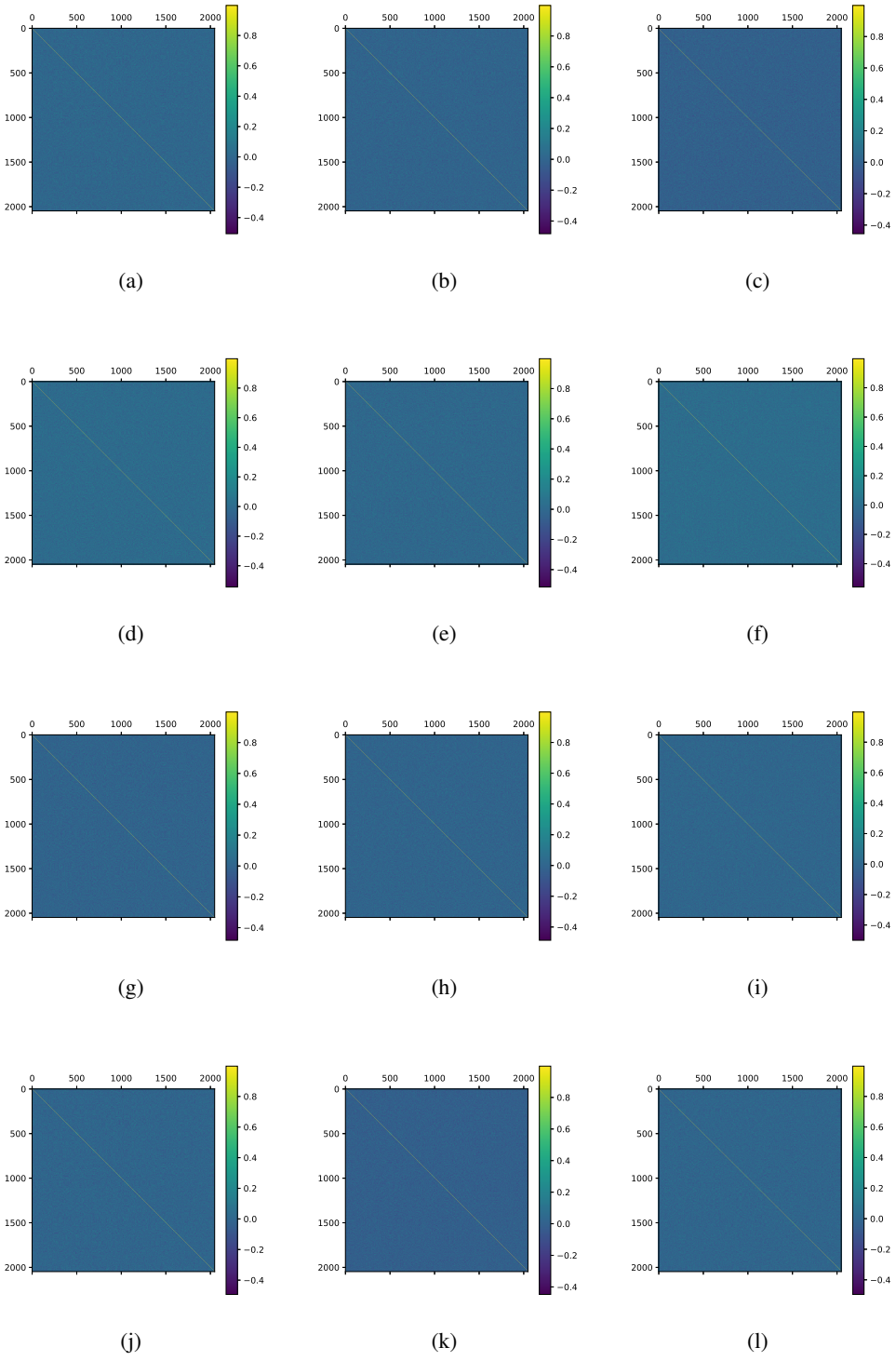


Figure 5: Visualization of the correlation matrix of SimSiam representations with 2048 output dimensions trained in different batches on 600 epoch. The correlation matrixes are close to identity matrix.